Simple Automated File Transfers Using SUP and Shift

April 4th, 2012

NASA Advanced Supercomputing Division

Overview

- SUP and Shift summary
  - What are they?
  - Why should you use them?
- Details of each
  - Usage
  - Features
  - Performance

Question? Use the Webex chat facility to ask the Host

Secure Unattended Proxy (SUP)

- What is it?
  - Authentication and authorization mechanism
  - Allows transfers and other remote commands to be invoked on NAS HEC systems without the use of SecurID for up to a week
- Why should you use it?
  - Highest performance transfer mechanism
    - Direct remote transfers to/from Pleiades, Columbia, and Lou
    - 10 GE connectivity with no intermediate disk limitations/bottlenecks
  - Transfers without the use of SecurID for a week
    - Scripted transfers are easily achievable
    - Interactive transfers are more convenient
  - ...and more

Question? Use the Webex chat facility to ask the Host

Self-Healing Independent File Transfer (Shift)

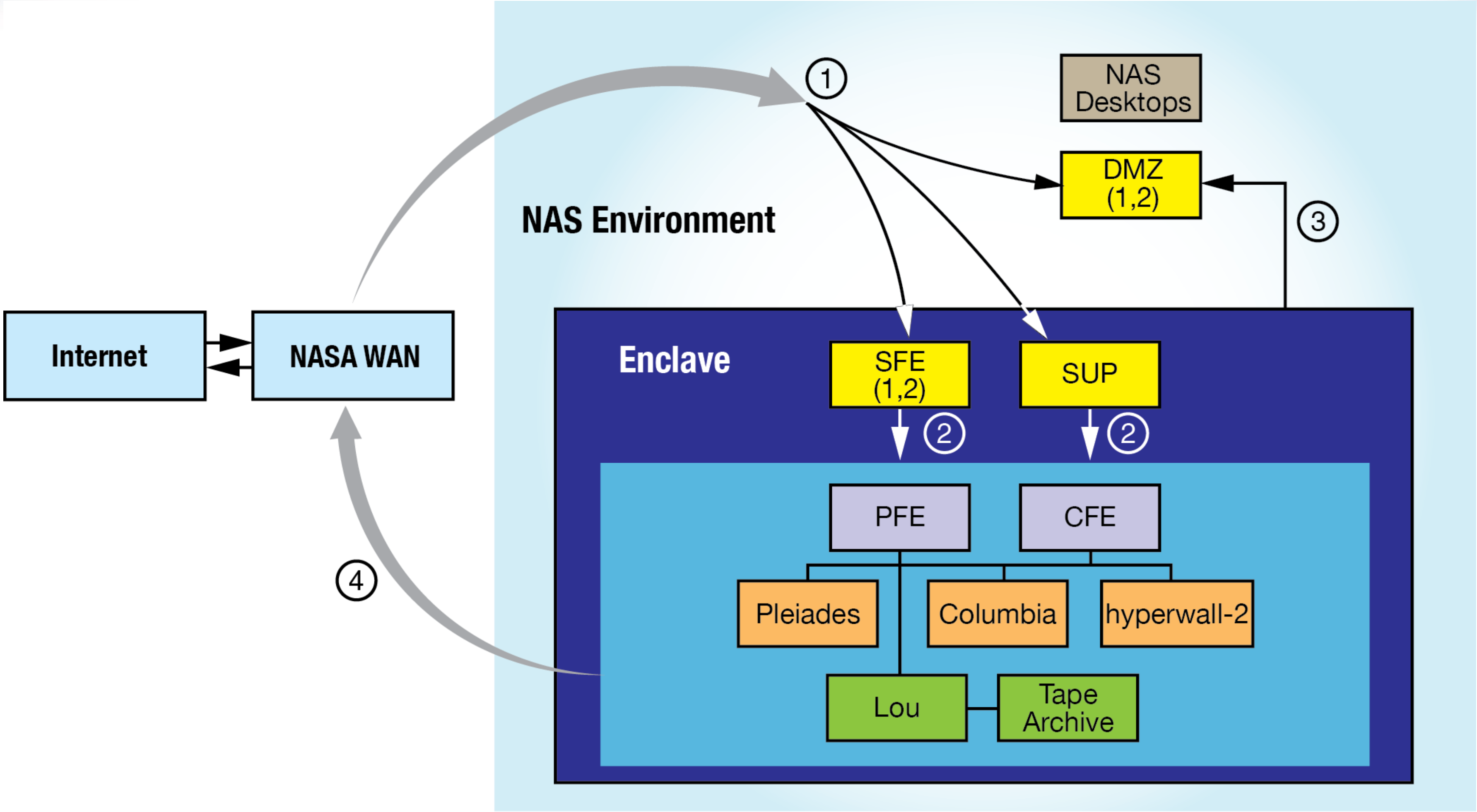
- What is it?
  - Automated file transfer mechanism
  - Built on top of SUP
- Why should you use it?
  - Supports local, intra-enclave, and remote transfers
  - As easy to use as cp/scp
  - Provides notifications and on-demand status of transfers
  - Takes care of numerous details so you don't have to
  - Advanced performance and reliability features

Question? Use the Webex chat facility to ask the Host

Secure Unattended Proxy (SUP)

Part 1/2

SUP Position Within NAS HEC Environment



🟨 Bastions ⬜ Front Ends 🟧 HEC Systems 🟩 Archival 🟫 Local Staff Systems

Question? Use the Webex chat facility to ask the Host

Basic SUP Model

- Obtain a "SUP key" using SecurID authentication
  - SUP keys are SSH private keys
  - SUP keys are valid for one week
- Use SUP key to invoke remote SSH commands on NAS HEC hosts
  - Currently authorized commands
    - bbftp (+bbscp), qstat, rsync, scp, sftp, test
    - bbcp and gridftp likely in near future
    - Other commands as needed
  - Currently supported hosts
    - pfe*, bridge*, cfe*, lou*, susan
- SUP operations always initiated from non-HEC host
  - Files can still be moved in any direction
- Remote SSH commands always hop through SUP system
  - Data can go direct to/from destination (discussed later)

Question? Use the Webex chat facility to ask the Host

SUP Prerequisites On Your Local Host

- SUP assumes availability/use of OpenSSH
  - Standard on most Linux/Unix distributions
  - Available on Windows via Cygwin or coLinux
- SUP client significantly simplifies SUP usage
  - Requires Perl >= 5.6.1 (Perl >= 5.8.5 for advanced features)
    - Standard on most Linux/Unix distributions
    - Available on Windows via Cygwin or coLinux
- SUP can be used without SUP client
  - Won't be discussed further
    - See http://www.nas.nasa.gov/hecc/support/kb/entry/241

Question? Use the Webex chat facility to ask the Host

SUP Client Setup On Your Local Host

1. Download client
   - Via browser
     - http://www.nas.nasa.gov/hecc/support/kb/file/9
     - Save downloaded file as "sup"
   - Via command-line tool
     - wget -O sup http://www.nas.nasa.gov/hecc/support/kb/file/9
     - curl -o sup http://www.nas.nasa.gov/hecc/support/kb/file/9
2. Make client executable
   - chmod 700 sup
3. Move client to directory in your $PATH
   - mv sup ~/bin

Question? Use the Webex chat facility to ask the Host

SUP Authorizations

- The SUP enforces restrictions not found in other transfer mechanisms since it permits direct transfers without SecurID
  - SUP functionality must be explicitly enabled
    - ~/.meshrc must exist on each NAS HEC enclave system
  - Only specifically authorized commands can be invoked on NAS HEC systems
    - Currently bbftp (+bbscp), qstat, rsync, scp, sftp, test
  - Only files in directories explicitly authorized by the user can be written
  - Dot files in the user’s home directory (~/.*) can never be read or written
  - You cannot interactively log in to the SUP
    - "ssh sup" = “Permission denied (unauthorized command)”
  - You cannot transfer files to the SUP itself
    - "scp /some/file sup:" = "Permission denied (unauthorized command)"

Question? Use the Webex chat facility to ask the Host

Authorization Setup On NAS HEC Hosts

- Authorize NAS HEC host for SUP operations
  - Create ~/.meshrc on pfe, cfe2, and/or lou1/2
- Authorize directories for writes
  - Add directories (one per line) to ~/.meshrc
  - A directory in ~/.meshrc allow writes to items in that directory and in any subdirectory
  - For example, /nobackup/user in ~/.meshrc implies:
    - /nobackup/user not removable
    - /nobackup/user/dir creatable/removable
    - /nobackup/user/file creatable/writable/removable
    - /nobackup/user/dir/.../file creatable/writable/removable

Question? Use the Webex chat facility to ask the Host

Using the SUP Client

- Begin command with "sup" (i.e. the SUP client)
- Run commands as if directly connected to NAS HEC resources
  - Client transparently rewrites command to flow through SUP
- Examples (from your local host)
  - sup scp /some/file pfe:/dir
  - sup bbftp -e "put /some/file /dir" pfe1.nas.nasa.gov
  - sup bbscp /some/file pfe1.nas.nasa.gov:
  - sup ssh pfe qstat
  - sup rsync /some/file pfe:/dir
  - sup ssh pfe test -f /dir/file
- Client may request information when new SUP key needed

Question? Use the Webex chat facility to ask the Host

Information That SUP Client May Request During SUP Key Generation

- Host key verification
  - Continue connecting if two printed SSH key fingerprints match
    - Example fingerprint: 52:f3:61:9b:9c:73:79:4d:22:cb:f3:cd:9a:29:4e:fe
- Authentication credentials
  - Passphrase for new/existing SSH private key (~/.ssh/id_rsa)
  - Password and SecurID passcode for {sup,sup-key}.nas.nasa.gov
- Client upgrade (please always answer "y"!)
  - Primary mechanism for distributing bug fixes and new features
    - Website version is mainly for initial setup
  - Client is regression tested before release
    - What worked before upgrade should work after upgrade
    - Some undetected problems on older SSH clients during Shift rollout
- Use client's -b option when running scripted SUP operations
  - Sets batch mode where user interaction disabled
  - Note that SUP operations will fail with -b if your SUP key has expired

Question? Use the Webex chat facility to ask the Host

SUP Client Sample Session

- When your SUP key has expired after a week

```
yourhost% sup scp /some/file pfe:dir
Generating key on sup.nas.nasa.gov (provide login information)
Password: ********
Enter PASSCODE: ********
Identity added: /home/user/.ssh/meshkey.1332973211
        (/home/user/.ssh/meshkey.1332973211)
Lifetime set to 604800 seconds
file 100% 3906MB 71.0MB/s 00:55
```

- When you already have a valid SUP key

```
yourhost% sup scp /some/file pfe:dir
file 100% 3906MB 71.0MB/s 00:55
```

- When you are a completely new SUP user
  - See http://www.nas.nasa.gov/hecc/support/kb/entry/145

SUP Client Caveats (User Names)

- Client does not currently support user names embedded within commands
  - sup scp /some/file user@pfe:/dir ✖
  - sup ssh -l user pfe qstat ✖
  - sup bbftp -u user -e "put /some/file /dir" pfe1.nas.nas.gov ✖
- If your local user name differs from your NAS user name, there are two options on your local host
  - Modify ~/.ssh/config to use NAS user name on sup/sup-key

    ```
    Host sup.nas.nasa.gov sup-key.nas.nasa.gov ✔
      User NAS_username
    ```

  - Use client's -u option
    - sup -u NAS_username scp /some/file pfe:/dir ✔

SUP Client Caveats (Commands)

- Bbftp (and bbscp since it uses bbftp)
  - NAS host names must be fully qualified outside NAS domain
    - sup bbftp -e "put /some/file /dir" pfe1 ✖
    - sup bbftp -e "put /some/file /dir" pfe1.nas.nasa.gov ✔
  - Load balancer aliases "pfe" and "bridge" will not work
    - sup bbftp -e "put /some/file /dir" bridge.nas.nasa.gov ✖
    - sup bbftp -e "put /some/file /dir" bridge1.nas.nasa.gov ✔
  - Port range may be needed if your site restricts outbound port usage
    - By default, the range 50000-51000 is used
    - Ranges 5000-5011 and 5020-5022 also available
    - sup bbftp -E "bbftpd -e 5000:5011" ...

SUP Client Caveats (Commands cont.)

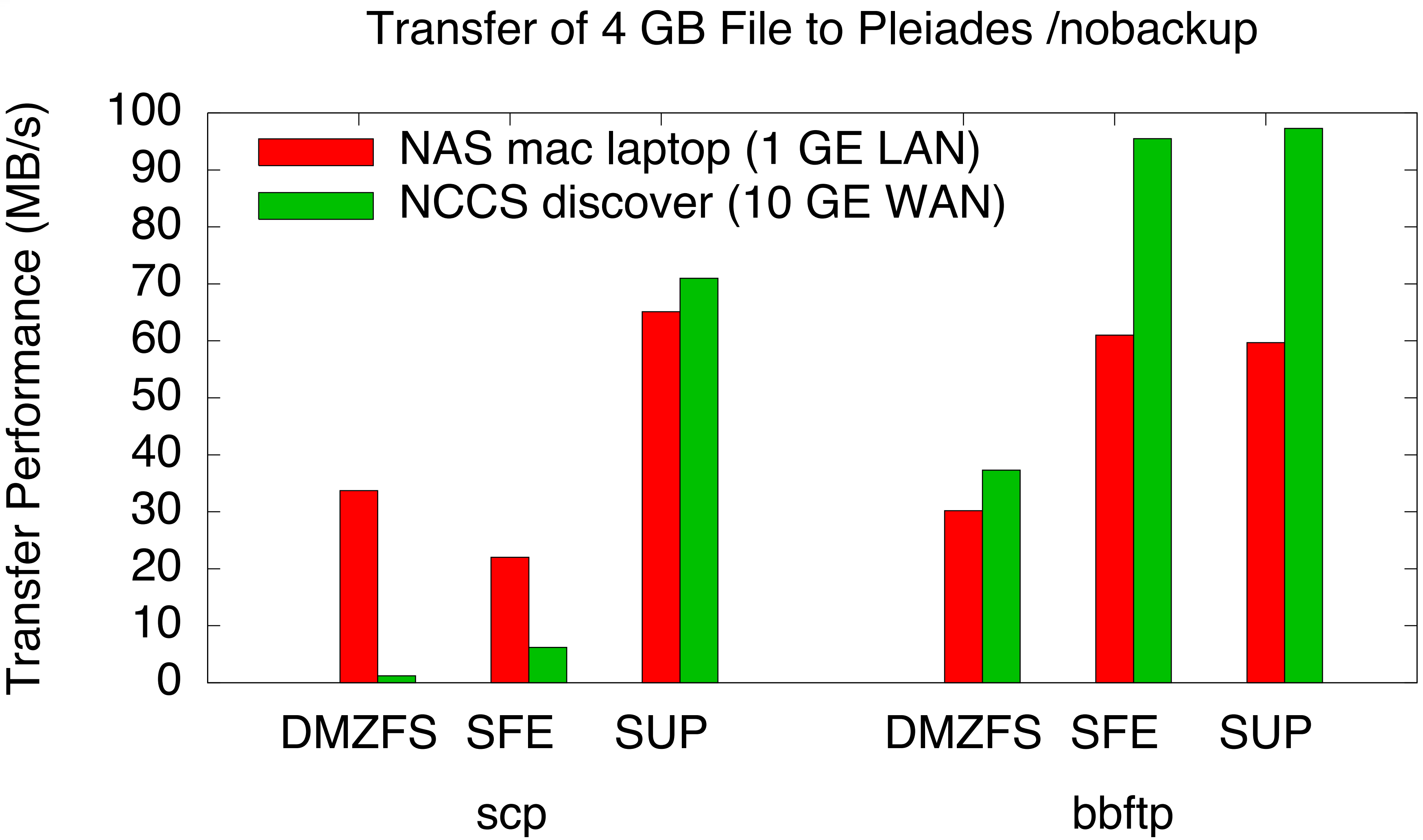
- Rsync
  - Transfers to home directory can fail even if home directory authorized for writes due to use of temp files starting with "."
    - sup rsync /some/file pfe: ✖
  - Use --inplace for rsync transfers to home directory
    - sup rsync --inplace /some/file pfe: ✔
- Scp
  - Third-party transfers are not supported
    - sup scp pfe:/some/file somehost:/dir ✖
    - scp pfe:/some/file somehost:/dir ✔ (SUP not involved)
    - ssh somehost sup scp pfe:/some/file /dir ✔
      - Assuming SUP client installed on somehost

Question? Use the Webex chat facility to ask the Host

SUP Troubleshooting

- Permission denied (~/.meshrc not found)
  - ~/.meshrc must exist on each NAS target host
  - Pleiades, Columbia, and Lou do not share home directories
- Permission denied (unauthorized command)
  - Only a specific set of commands is allowed through the SUP
  - Some specific options may be disabled
    - Not needed in common usage scenarios
- Permission denied during file access (various forms)
  - Destination directory has not been authorized for writes
    - Add destination directory or parent directory to ~/.meshrc
  - Attempted read/write of ~/.*

SUP Performance Comparison



Question? Use the Webex chat facility to ask the Host

Increasing SUP Transfer Performance

- Tune your local host(s) for WAN transfers
  - See previous webinar
    - "How Can I Speed Up My Data Transfers to/from NAS?"
  - See http://www.nas.nasa.gov/hecc/support/kb/entry/137
  - Note that WAN tuning may impact your LAN performance
- Install/use bbftp instead of rsync/scp/sftp
  - Bbftp data channels bypass SUP entirely
  - Bbftp data channels are not encrypted
    - Do not use bbftp for sensitive data!
  - See http://www.nas.nasa.gov/hecc/support/kb/entry/147

Question? Use the Webex chat facility to ask the Host

SUP Advanced Features

- Remote job monitoring using qstat for automated workflows
  - For example, transfer results when job done (in bash syntax)

```
while [ ! -d results ]; do
    sup -b ssh pfe qstat job.id || sup -b scp -r pfe:/dir/results .
    sleep 600
done
```

- Remote file monitoring using test for automated workflows
  - For example, transfer intermediate results of running job (in bash syntax)

```
while [ ! -d part1 ]; do
    sup -b ssh pfe test -f /dir/part1.done && sup -b scp -r /dir/part1 .
    sleep 600
done
```

  - Here, the job must create /dir/part1.done when results available

SUP Advanced Features (cont.)

- SUP virtual file system
  - Remotely access files on NAS HEC hosts using standard file system commands via bash shell
    - Remote files specified in scp HOST:/PATH format
  - Currently supported commands
    - cat, cd, chgrp, chmod, chown, cmp, cp, df, diff, du, file, grep, head, less, ln, ls, mkdir, more, mv, pwd, rm, rmdir, tail, tee, test, touch, wc
  - Enable with eval `sup -s bash` and disable with eval `sup -r bash`
  - Why use it?
    - Intersperse local/remote file operations on command line and in scripts
      - Scripts must use "#!/bin/bash" and start with "shopt -s expand_aliases"
    - Most efficient way to repeatedly transfer and/or operate on small files
      - Uses a single connection for all file operations on the same host
  - Functionality has various caveats and limitations
    - Avoid "whole file" commands (cat, cmp, diff, grep, wc) on large files!
    - See http://www.nas.nasa.gov/hecc/support/kb/entry/240

Question? Use the Webex chat facility to ask the Host

SUP Virtual File System Sample Session

```
yourhost% bash
bash-3.2$ export PS1="\h[\w]> "
yourhost[~]> eval `sup -s bash`
yourhost[~]> cd pfe:/etc
yourhost[pfe:/etc]> ls -l HOST<tab>
yourhost[pfe:/etc]> ls -l HOSTNAME
-rw-r--r-- 1 root root 18 Apr 05 2011 /etc/HOSTNAME
yourhost[pfe:/etc]> diff HOSTNAME /etc/HOSTNAME
1c1
< pfe5.nas.nasa.gov
---
> yourhost.yourdomain
yourhost[pfe:/etc]> cp HOSTNAME ~
yourhost[pfe:/etc]> cat ~/HOSTNAME
pfe5.nas.nasa.gov
```

From SUP To Shift

- SUP provides benefits on its own
  - Improved performance
  - Increased convenience
  - Scripted transfer and file/job monitoring
  - Virtual file system
- Users still responsible for understanding transports
  - Which transports are available
  - When and how to use each transport
  - What idiosyncrasies may exist in each transport
  - How to optimize each transport
  - How to detect and recover from failures in each transport

Question? Use the Webex chat facility to ask the Host

From SUP To Shift (cont.)

- Users still responsible for understanding hosts
  - Which hosts are available
  - Which hosts likely to have best performance in general
  - Which hosts likely to have best performance at a given time
- Shift extends SUP functionality with an automated transfer capability
  - Shift has extensive embedded knowledge of transports
  - Shift has embedded knowledge of NAS hosts
  - Shift is integrated with load balancing infrastructure

Question? Use the Webex chat facility to ask the Host

Self-Healing Independent File Transfer (Shift)

Part 2/2

Shift Transfers

- Simple
  - Drop-in replacement for cp/scp
- Automated
  - Fire and forget with notifications of completion, errors, and warnings by email
  - On-demand status
- Reliable
  - Recovers from multiple types of system/transport failures
  - Restart of failed transfers
  - Optional integrity verification with partial file retransmission
- Fast
  - Automatic transport selection and optimization
  - Automatic host selection based on availability, load, and performance
  - Integrated DMF management for optimized file recall
  - Single and multi-file parallelization

Question? Use the Webex chat facility to ask the Host

Shift Client Setup

- NAS HEC hosts
  - None!
  - Already exists as "shiftc" in /usr/local/bin
- Remote hosts
  - Install SUP client if not already done
  - Embedded within client using "sup shiftc ..."
  - Still need to authorize NAS HEC hosts for SUP operations
    - Create ~/.meshrc if it does not exist
  - Still need to authorize directories for writes
    - Add top level directories to ~/.meshrc

Question? Use the Webex chat facility to ask the Host

Shift Transfer Initialization

- Local transfers (just like "cp")
  - bridge% cp /file1 /file2
  - bridge% shiftc /file1 /file2
- Intra-enclave transfers (just like "scp")
  - bridge% scp /file1 lou:/file2
  - bridge% shiftc /file1 lou:/file2
- Remote transfers (just like "sup scp")
  - yourhost% sup scp /file1 pfe:/file2
  - yourhost% sup shiftc /file1 pfe:/file2
  - SUP user name caveats still apply
  - Can use pfe/bridge aliases and unqualified NAS hosts from anywhere!

Question? Use the Webex chat facility to ask the Host

Shift Transfer Initialization (cont.)

- Common initialization options
  - Recursive transfers (-r, -R, --recursive...just like cp/scp)
    - Copy directories recursively
  - Attribute preservation (-p, --preserve...just like cp/scp)
    - Preserve times, permissions, and ownership
  - Link preservation (-P, --no-dereference...just like cp)
    - Never follow symbolic links
    - Can result in broken links at destination
  - Link dereferencing (-L, --dereference...just like cp)
    - Always follow symbolic links
    - Can result in duplicate files at destination
  - Data encryption (--encrypt)
    - Encrypt data stream(s) during remote transfers
    - Eliminates bbftp so may reduce performance

Question? Use the Webex chat facility to ask the Host

Shift Transfer Initialization (cont.)

- Shift computes file operations in transfer and prints transfer id
  - Initialization output

    ```
    Directories/files found: 0/1
    Shift id is 1
    ```

  - The id can be used to manage/obtain status about a particular transfer
- So far, Shift looks like any other transfer command
  - What are the benefits over traditional transfers...?

Shift Benefit: Background Transfers

- After initialization, Shift detaches and begins the transfer
  - You do not need to stay logged on to the origin system
  - You will be notified by email of completion, errors, and/or warnings
- A running transfer may be stopped at any time from any host
  - shiftc --stop --id=N
  - Batches of file operations in progress will run to completion
- Shift provides history of transfers
  - shiftc --history
  - Transfer data only kept for one week after completion/error/stop!

Question? Use the Webex chat facility to ask the Host

Shift Transfer History

```
pfe% shiftc --history
```

```
id | origin            | command
---+-------------------+----------------------------------------------------
 1 | pfe1.nas.nasa.gov | shiftc file1 /tmp/dir1
 2 | pfe1.nas.nasa.gov | shiftc -p file1 cfe2:
 3 | your_localhost    | sup shiftc -r --verify /tmp/dir1 cfe2:/tmp/dir2
 4 | your_localhost    | sup shiftc -r --encrypt cfe2:/tmp/dir2 .
 5 | pfe1.nas.nasa.gov | shiftc -r --hosts=4 bigdir1 /nobackup/user1/bigdir2
```

Shift Benefit: Transfer Status

- Traditional transfers often provide minimal feedback
  - cp: no output
  - scp: long scrolling list of file names without directories
- Users left wondering about running transfers
  - What has been done?
  - What is left to do?
  - Are there any problems?
  - How long is it likely to take?
- Shift provides summarized status of all transfers
  - State, portion complete, and performance
  - shiftc --status
- Shift provides detailed status of individual transfers
  - State, error messages, tool used, and performance of each file operation
  - shiftc --status --id=N

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Transfer Status (cont.)

```
pfe% shiftc --status

id | state | dirs | files |      file size | start | time |    rate
   |       | sums | attrs |       sum size |       |      |
---+-------+------+-------+----------------+-------+------+---------
 1 | done  |  0/0 |   1/1 |      92KB/92KB | 10/03 |   2s |   46KB/s
   |       |  0/0 |   0/0 |      0.0B/0.0B | 17:06 |      |
 2 | done  |  0/0 |   1/1 |      92KB/92KB | 10/03 |   8s | 11.5KB/s
   |       |  0/0 |   1/1 |      0.0B/0.0B | 17:06 |      |
 3 | done  |  1/1 |   2/2 |      99KB/99KB | 10/03 |   1s |   99KB/s
   |       |  4/4 |   0/0 |    198KB/198KB | 17:07 |      |
 4 | error |  1/1 |   1/2 |      92KB/99KB | 10/03 |   3s | 30.7KB/s
   |       |  0/0 |   0/0 |      0.0B/0.0B | 17:08 |      |
 5 | done  |  1/1 | 64/64 |  65.5GB/65.5GB | 10/03 |  29s | 2.26GB/s
   |       |  0/0 |   0/0 |      0.0B/0.0B | 17:09 |      |
```

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Transfer Status (cont.)

```
yourhost% sup shiftc --status --id=2

state | op     | target              | size | start | time |   rate
      | tool   | message             |      |       |      |
------+--------+---------------------+------+-------+------+-------
done  | cp     | cfe2:/u/user1/file1 | 92KB | 10/03 |   5s | 18KB/s
      | bbftp  | -                   |      | 17:06 |      |
done  | chattr | cfe2:/u/user1/file1 |    - | 10/03 |   1s |      -
      | sftp   | -                   |      | 17:06 |      |
```

```
yourhost% sup shiftc --status --id=4 --state=error

state | op    | target             | size | start | time | rate
      | tool  | message            |      |       |      |
------+-------+--------------------+------+-------+------+-----
error | cp    | /tmp/dir2/file2    |  7KB |     - |    - |    -
      | rsync | rsync: send_files  |      |       |      |
      |       | failed to open:    |      |       |      |
      |       | Permission denied  |      |       |      |
```

Shift Benefit: Transport Selection

- In traditional transfers, users are left deciding which transport should be utilized and how
  - May be better transports available
  - May be unknown options that increase performance
  - May be unknown idiosyncrasies
- Shift chooses most effective transport for each task
  - Availability (use transports available at both source and destination)
  - Performance (use higher performance transports first)
  - Functionality (e.g. bbftp cannot perform partial transfers)
- Shift understands how to utilize selected transport
  - How to construct command lines
  - Which arguments to use for optimum performance
  - How to parse output to detect success or failure

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Transport Selection (cont.)

- Currently supported transports (more to come)
  - Local: mcp (high performance cp), rsync, cp (built-in perl equivalent)
  - Remote: bbftp, rsync, sftp (built-in perl equivalent)
- Can still force specific transport(s)
  - --local={mcp,rsync,cp} or --remote={bbftp,rsync,sftp}
    - shiftc --local=mcp,cp /some/file /dir
    - shiftc --remote=rsync /some/file lou:/dir
  - Only recommended for specific scenarios
    - Force use of rsync to synchronize mostly similar directories
    - Test performance/functionality of specific transport
    - Repeated errors with default Shift selection (notify NAS support!)
  - Shift will always use built-in transports (cp/sftp) for certain tasks

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Host Selection

- Original host(s) specified in transfer may be non-optimal
  - May become unavailable during large transfers
  - Other activity on system may degrade performance
  - Higher performance options may be available
    - For example, system with 10 GE interface
- Shift automatically replaces remote NAS hosts with lower load and/or higher performance systems
  - Any host used will have equivalent file system access
    - File operations will be rewritten as necessary
  - Original remote host given will likely not be used
  - Host on which transfer initiated will always be used

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Transfer Recovery and Restart

- Failures during traditional transfers can waste significant time
  - Manually determine which parts of the transfer have failed and manually construct the commands to retry them
  - Redo the entire transfer without regard to previously transferred files
- Shift tracks and classifies failures by recoverability
  - "Recoverable" errors will be attempted again automatically
  - "Unrecoverable" errors will not be retried
  - Shift currently classifies more as unrecoverable than recoverable
    - Classifications will be refined over time
    - In the meantime, use restart capability...

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Transfer Recovery and Restart (cont.)

- Shift allows failed/stopped transfers to be easily restarted
  - Completed operations will not be run again
  - Failed operations will be retried
  - Operations that were never attempted will be performed
- Restarting a transfer with a given id
  - shiftc --restart --id=N
  - Transfer has previously failed with errors or been stopped
  - Must restart on original host or one with equivalent file system access!
- Shift automatically resumes transfers after system reboots or process failures
  - Shift inserts a crontab entry for each running transfer
  - Cron-invoked Shift process will check on health of initial Shift process
    - Will take over as needed
  - Crontab entries will be cleaned up upon transfer completion

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Integrity Verification

- Files traverse many components during a transfer
  - Subject to transient failures within each component
  - May induce corruption not detectable by error detection/ correction measures of each
- Traditional hash (e.g. md5sum) verification only indicates that some part of the file has changed
  - Either manually find difference or transfer again
    - Both are time consuming for large files

Question? Use the Webex chat facility to ask the Host

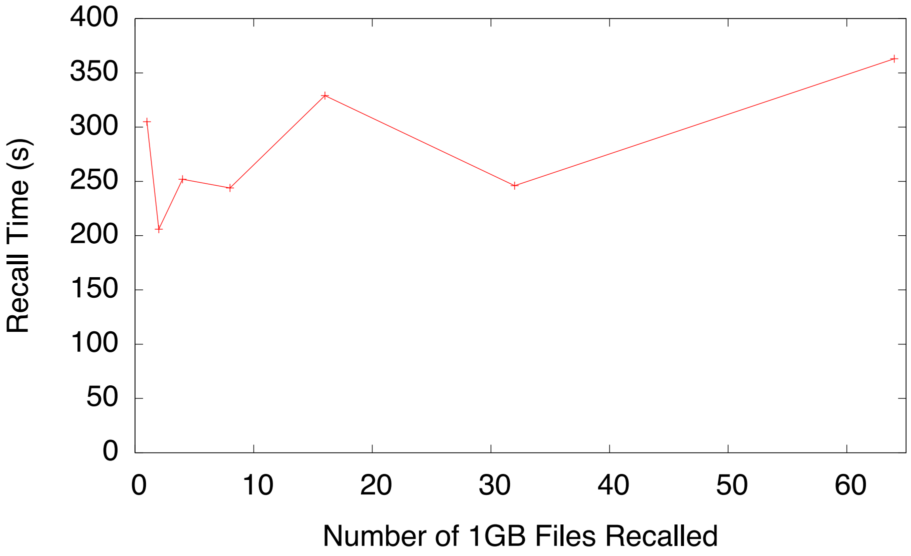
Shift Benefit: Integrity Verification (cont.)

- Shift can verify that file contents on destination disk match contents on source disk using --verify
  - shiftc --verify /some/file pfe:/dir
- Shift can locate the source of corruption
  - Currently detects corruption at granularity of 1 GB
- Shift can rectify corruption using partial transfers
  - Currently rectifies corruption at granularity of 1 GB
- Computationally expensive so optional
  - Higher impact on high speed links

Question? Use the Webex chat facility to ask the Host

Shift Benefit: DMF Management

- Files on Lou may have been migrated to tape
- Migrated files must be brought online before transfer
- DMF does this for you during transfers from Lou
  - Can be extremely slow as files recalled one at a time



Question? Use the Webex chat facility to ask the Host

Shift Benefit: DMF Management (cont.)

- Should use dmget to recall files before transfer
  - Lou has a limited amount of online storage
  - Large amounts of data may need to be recalled in stages
- Shift does dmget for you
  - DMF file systems automatically detected
  - Migrated files are recalled in batches as needed
- Shift does dmput for everyone
  - Files migrated to tape after successful transfer/verification
  - Preserves online storage resources for other users
  - Use --no-offline to prevent migration
    - shiftc --no-offline /some/file lou:
    - shiftc --no-offline lou:/some/file /dir

Question? Use the Webex chat facility to ask the Host

Shift Benefit: Multi-File Parallelization

- The maximum transfer rate between two sites is often greater than that achievable between two hosts at those sites
  - Single host limitations (I/O rate, network interface, CPU, ...)
- Shift can use multiple hosts to carry out the same transfer with the --hosts option
  - Must be multiple sources/targets with access to same file system
    - Local transfers on Pleiades
      - `shiftc --hosts=4 -r /nobackupp3/user/dir /nobackupp4/user/dir`
    - Remote transfers from other clusters to/from Pleiades
      - `sup shiftc --hosts=2 -r /some/dir pfe:/nobackup/user/dir`
  - No effect otherwise
    - Transfers from Lou/Columbia home file systems
    - Remote transfers from standalone systems

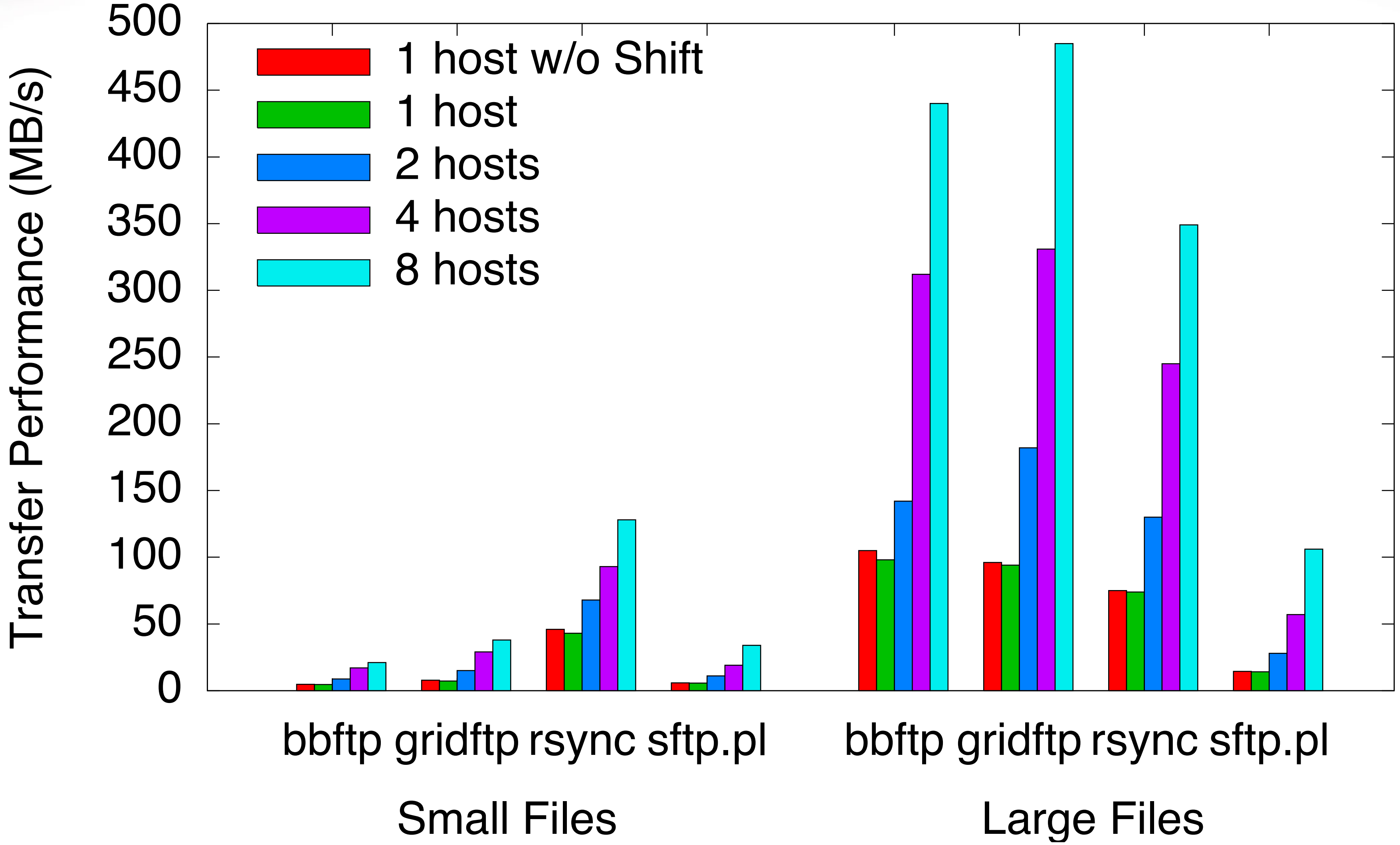
Question? Use the Webex chat facility to ask the Host

Shift Benefit: Multi-File Parallelization (cont.)

- Shift knows about hosts and file systems within the NAS HEC environment
- Shift does not (initially) know about hosts and file systems within your environment
  - Shift learns incrementally when you initiate transfers
    - Knows you have access to origin host
    - Knows which shared file systems exist on origin host
    - Stores this information in ~/.shift.fs on any host that initiates a transfer
  - You can populate this info more quickly using a small transfer from each host
- Information used by Shift to spawn clients on other hosts
  - You have access to the host
  - The host has access to the relevant local file system
  - There is enough work left for the host
  - You can authenticate to the host from the origin host
    - Hostbased authentication
    - Pubkey authentication with SSH agent running on origin host

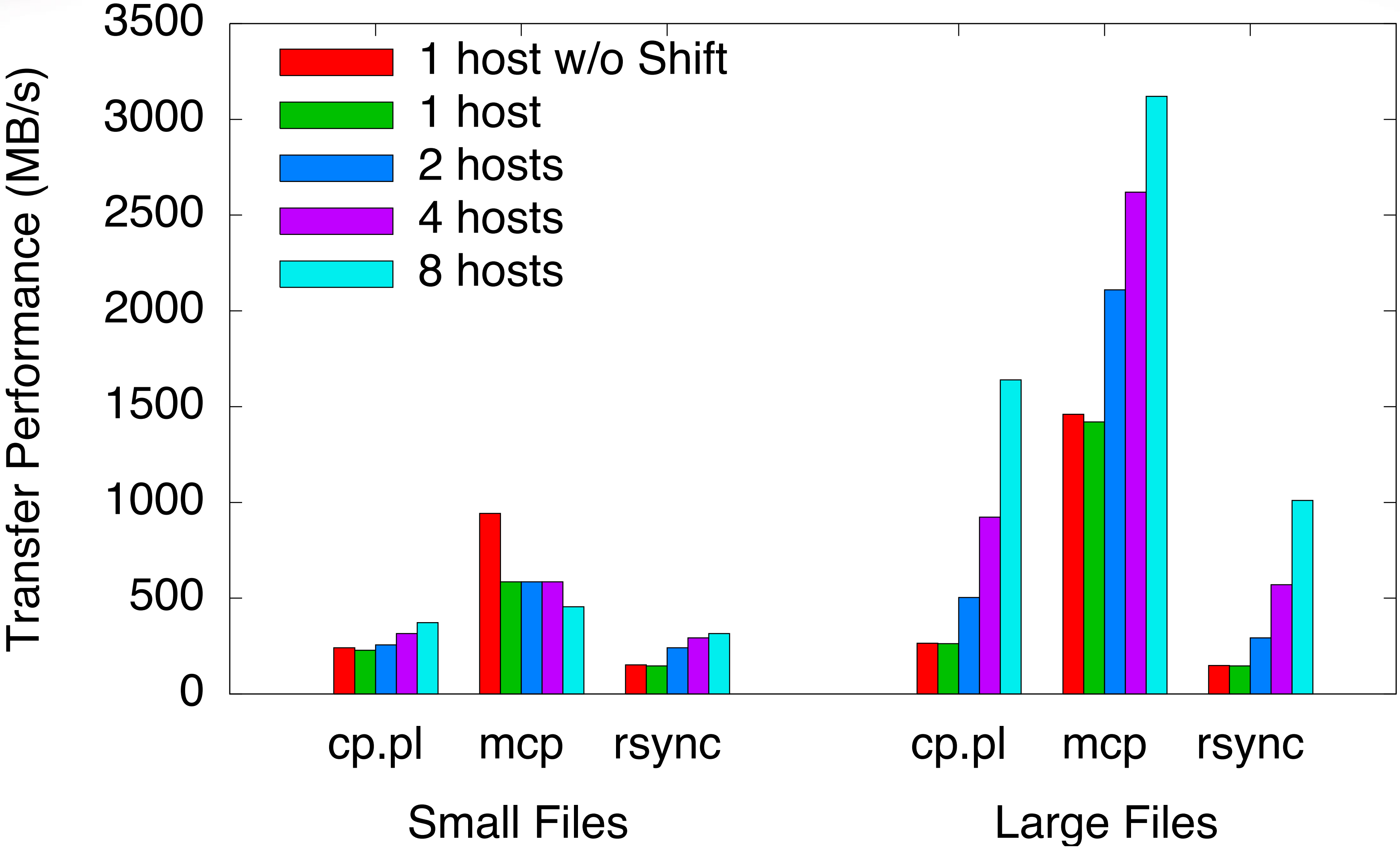
Question? Use the Webex chat facility to ask the Host

Shift Remote Performance
(1k*4MB Files and 64*1GB Files via 10 GE WAN)



Question? Use the Webex chat facility to ask the Host

Shift Local Performance (1k*4MB Files and 64*1GB Files Lustre->Lustre)



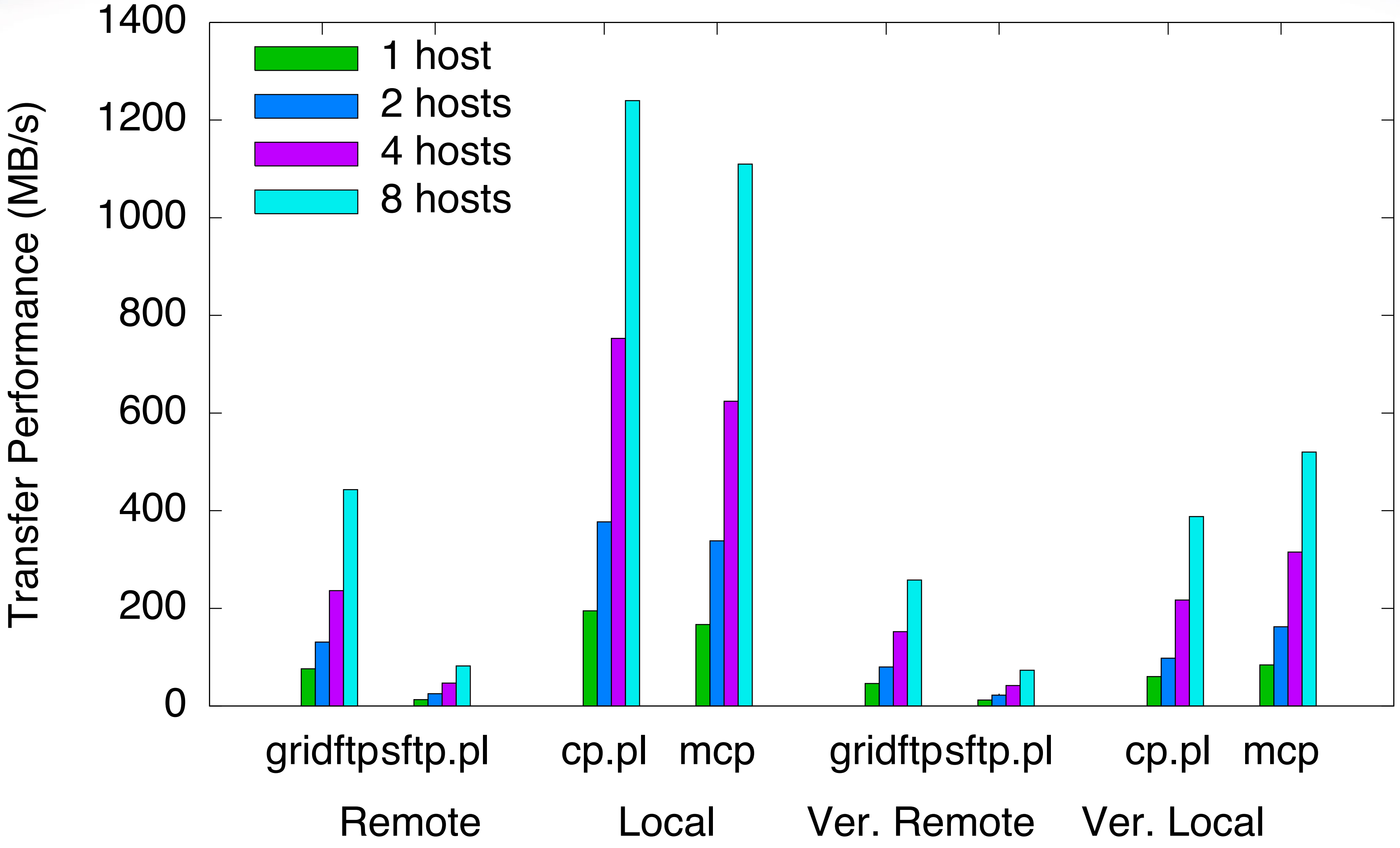
Question? Use the Webex chat facility to ask the Host

Shift Benefit: Single File Parallelization

- Transfers of large files can cause parallel clients to become imbalanced
  - One client transferring file, others remain idle
- Shift supports parallelization of single file transfers using --split-size option
  - sup shiftc --split-size=1g --hosts=8 /big/file bridge:/dir
  - No effect unless --hosts also specified
- Uses partial transfer capabilities of some transports
  - Local case uses mcp or built-in cp
  - Remote case uses gridftp or built-in sftp
    - Problem 1: gridftp not deployed for all users yet
    - Problem 2: sftp is slow
    - Solution 1: gridftp will be deployed for all
    - Solution 2: built-in sftp will be replaced with faster option

Question? Use the Webex chat facility to ask the Host

Shift Single File Parallelization Performance (1 64GB File via 10 Gb/s WAN and Lustre->Lustre)



Question? Use the Webex chat facility to ask the Host

Some Initial Problems (Now Fixed)

- SUP/Shift client used ciphers/MACs not available in some older OpenSSH clients
  - Client now tests for existence of ciphers/MACs before use
- Shift selected pfe's for lou transfers instead of bridge nodes
  - Resulted in saturation of 1 GE pfe links and bbftp errors
  - Shift now uses bridge nodes to/from Lou when available
  - If you transfer to/from Lou, initialize transfer from bridge/lou!
- High performance patch of SSH client on newly deployed 10 GE SUP could exceed max TCP buffer size
  - Resulted in lost connections in some scenarios
  - Replaced with stock SSH client

Question? Use the Webex chat facility to ask the Host

Conclusion

- Secure Unattended Proxy (SUP)
  - Functionality
    - Interactive/scripted transfers and remote file/job monitoring
  - Convenience
    - Only need to use SecurID once a week
    - Simple to use by prepending "sup" to commands
      - "scp /some/file pfe:" becomes "sup scp /some/file pfe:"
  - Performance
    - Direct transfers to NAS HEC hosts over a high speed link with no intermediate storage limitations/bottlenecks
  - Knowledge base article
    - http://www.nas.nasa.gov/hecc/support/kb/entry/145

Question? Use the Webex chat facility to ask the Host

Conclusion (cont.)

- Self-Healing Independent File Transfer (Shift)
  - Functionality
    - Automated file transfer with advanced tracking and failure recovery
  - Convenience
    - Fire and forget transfers with simple usage a la cp/scp
  - Performance
    - Numerous optimizations spanning transports, hosts, and environments
    - Parallelization can achieve rates far beyond standard transports
  - Some near term enhancements, among others...
    - Support for bbcp and gridftp with transport selection based on file sizes
    - Significantly faster built-in remote transport
      - Early prototype over 6x faster than sftp in initial testing
      - Will make single file parallelization more practical without gridftp
  - Knowledge base article
    - http://www.nas.nasa.gov/hecc/support/kb/entry/300

Question? Use the Webex chat facility to ask the Host